SAIRO: Semantic Association based Information Retrieval using Ontology

1Sakthi Murugan R, 2P. Shanthi Bala, 3Dr. G. Aghila
1, 2 Department of Computer Science, Pondicherry University, Puducherry, India
3 Department of Computer Science & Engineering, National Institute of Technology (NIT), Karaikal, Puducherry, India

Abstract— This paper presents a technique to expand the user query in information retrieval using semantic association rules from Ontology. Though a number of ways to semantically expand the user query have been proposed, the association rules that are related to concepts in Ontology is not explored well. SAIRO takes full advantage of association rules to semantically understand and expand the user query with the help of domain Ontology. These semantically expanded query by the SAIRO engine is used to retrieve the relevant documents. SAIRO system have been modelled and designed in a generic way, so it can be integrated with the existing search engine and also facilitates to add new domain knowledge easily. A collection of more than 7400 documents related to computer networks domain and computer network Ontology have been used to evaluate SAIRO. Time complexity of SAIRO is computed to be $O(1)$ in best case and $O(C+R)$ in worst case. An average of 8.8% improvement in precision $P@10$ with 0.01 second increase in processing time have been achieved.

Keywords— Ontology, Information retrieval, Semantic association, Query Expansion.

I. INTRODUCTION

Internet evolution added greater convenience to share knowledge through World Wide Web (WWW). But most of the users find inconvenience in retrieving the relevant information from web. Search engines mostly addresses this problem by using pattern-matching technique [1]. It helps to retrieve the documents that match with the given keyword from web. Enormous research work is carried out to incorporate intelligence in information retrieval system. Semantic web [2] views WWW as a collection of data instead of set of documents by incorporating intelligence in the existing web. This work employs semantic web technologies in retrieving more relevant information from web by understanding the user query with the help of Ontologies [3]. Ontologies represents collection of concepts with their relations between them, their instances in machine understandable and human readable format.

For example, a user searching for 'Travel to India', the system is also expected by user to return the modes of travel like 'Flights to India', 'Trains to India', or any possible mode of transportation to India. Here, the user expects the information retrieval system to understand the keywords given by the user. This is done by semantically expand the user query by adding new related terms, to retrieve more relevant documents.

Semantic search facility is also available in existing popular search engines like Google and Bing. Google's Hummingbird algorithm adds semantic features in search with the combination of the knowledge graph. Similarly many research works to expand the user queries semantically have been discussed in section 2.

For example, consider two concepts 'flash drive' and 'storage device'. The association between these concepts is 'kind of', which relates the concepts. Associations are additional source of intelligence as it provides the system with the relation between concepts in Ontology. In this example, it means 'flash drive a kind of storage device'. Semantic Association based Information Retrieval using Ontology (SAIRO) uses domain Ontology to understand the user query. SAIRO accepts user input in terms of concepts and associations. SAIRO engine is used to semantically expand the user query with the help of Ontology. It uses semantically expanded query to retrieve the relevant documents. Section 3 gives the detailed architecture and work flow of SAIRO

The SAIRO search engine has been developed as a web application with Java/ J2EE development technology. Initially, computer network Ontology is developed using Protégé tool. The concepts and association rules from the computer network Ontology are accessed through Java program. Eclipse is used to develop the web application and Tomcat is used to publish it. The overall work flow of Java classes and their interaction with Ontology is described in section 4 along with the implementation of test cases and the method to incorporate SAIRO engine to any existing search engine.

Search results are evaluated with precision and recall for top ten results. In Section 5, a comparative analysis of SAIRO have been made with existing search facility like Lucene, Google and Bing in terms of precision, recall, F-measure, time complexity and ratio of relevance in expanded terms. From the results, it is observed that the performance of SAIRO provides better accuracy as compared with other search engines. Section 6 briefly concludes the work being performed.
II. RELATED WORKS

Various types of information techniques like keyword based search, vector space model, probabilistic model, context based model and semantic based model have been discussed in [4]. The conventional information retrieval system retrieves documents based on pattern matching. This method does not fetch relevant documents unless the user knows the exact keyword as specified in the documents. To overcome this issue, many research have been performed to incorporate the semantic intelligence in search engine, which helps to fetch more relevant documents.

In vector space model [5], the documents are retrieved based on the cosine similarity between the documents and the keywords. In probabilistic model [6], the documents are retrieved based on the ratio of relevant portion of the document to the irrelevant portion of the documents. Context based model [7] had some impact over other models as it considers the user context of search such as user profile, location, time, etc. For example, the user has given the search term in context based model as 'dress shop', based on gender information from the user's profile and current location, the search suggest some nearby shops to buy cloths.

In general, semantic search is performed based on semantic similarity, semantic association and semantic annotation. Semantic similarity [8] is the method of adding similar terms from any lexical data source, which provided alternate words for a given word. Semantic association based information retrieval [9] analyse the relation between the user entered concepts to understand the user context of query to retrieve the relevant documents. In semantic annotation based information retrieval [10], the developer will annotate the webpage with all possible keywords through which the user will get into the webpage. It makes the implementation of search engine much simpler as it uses the tags as given by developers to index the documents. But it is not feasible to implement by annotating billions of existing web pages with semantic tags. It is also possible for irrelevant page to rank higher in search result by simply adding all the current popular tag words. Another module to filter this spam pages must also be implemented, as the tag words by the developer cannot be trusted.

In most of the models, query reformulation had a significant impact in fetching more relevant documents. This query reformulation is done by semantically adding relevant terms to the user query, Miriam et al. [11] proposed a semantic information retrieval framework using Ontology. In this model, the user feeds the query directly in the SPARQL editor. This SPARQL query is directly executed in the knowledge base and those documents that are indexed with the SPARQL query are fetched.

Min Song et al. [12] developed semantic query expansion using association rules. This model first fetches set of documents with the user query. Based on the user selection, the query is reformulated by analysing the association in the pages with the user input terms. A method of rough Ontology based information retrieval was proposed by HU Jun et al. [13] where the initial set of documents are retrieved using regular keyword based approach and an association search in Ontology is also made in parallel. Degree of similarity of elements is used in order to rank the documents.

Miao Chen [14] explored the use of relations in Ontology to retrieve relevant information where the user query terms are identified in Ontology and further their relations are used to expand the query. Payam Barnaghi et al. [15] reformulated semantic association discovery algorithm, where complex relations between entities is found by recursively finding all the paths between the Ontologies and finally ranking based on domain Ontology. Alejandra et al. [16] designed Ontology based query expansion to search for learning resource. Common Ontological relations, domain specific relations, traditional terminological relations are used in query expansion process with a lexical distance of one. If the terms don’t match, then closest term is used in retrieval of documents.

On August 2013 Google updated to Hummingbird [17] algorithm which makes the search more semantic way, where it understands the meaning behind the user query to retrieve the documents. Google incorporated this semantic search to analyse the user query and linking it to knowledge graph. Bing also have introduced adaptive search [18], where the system knows the user and the context of search to determine the search results. So, different user with the same keyword will get different results based on their previous activities.

There are lots of advantages in incorporating appropriate semantic in existing search feature. The search engine understands the user query by incorporating semantics of the given keyword in the search. Search engines can perform quite remarkable by getting the user directly what they need. In the analysis, it is found that the semantic association based model intend to retrieve more relevant documents. The following work utilizes the association rules extensively from the Ontology to retrieve more relevant documents.

III. SAIRO

The retrieval of most relevant information from the web is a tedious task for naive users. It is due to the expectation of the search engine from the user, to enter the exact terms as specified in the documents. In most of the cases, it is difficult for the user to feed the search engines with the terms that exactly matches with the relevant documents. SAIRO manages to retrieve more relevant documents even though the keyword given by user did not exactly match with the documents. This is achieved by semantically expanding the user given terms with the help of Ontology.

A. Ontology

Ontology is the representation of knowledge in Web Ontology Language (OWL) [19] and Resource Description Framework (RDF) [20] formats. SAIRO uses computer network domain Ontology [21], which is the collection of concepts and relations between the concepts. Figure 1 shows snapshot of part of computer network Ontology.
B. Architecture

Initially the web is crawled with feed URL's to prepare the index for SAIRO search engine. The hyperlinks in the crawled pages are scanned and recursively added to the list to process the pages. Inverted index technique is used to index the web documents to provide fast full text search. Documents are mapped to the keyword using inverted index technique. Even though inverted index technique consumes high pre-processing time, it is being preferred as it performs search much faster than normal indexing. Crawling and indexing must be performed at regular time interval, in order to get the latest information from webpages, as webpages like news sites update their page more often. Once crawled and indexed, the user is ready to use the search features by directly entering the query or the search terms. Figure 2 shows the architecture of SAIRO based information retrieval.

C. SAIRO Engine

The SAIRO engine first analyses the user given terms by comparing it with the absorbed knowledge from the computer network Ontology in order to determine the user-given term is a concept or a relation. Figure 3 shows the flow diagram of SAIRO engine operations, where \( Cu, Ru \) represent user given concepts and relations respectively. B.

1) Single Concept Match: If a single user given term matches with the concept in Ontology, then all the neighbouring concepts along with the associations are used in retrieving the documents. Considering the Ontology as a directed graph with all the concepts as nodes and the relations as links can represent single concept match. Two nodes in a graph are said to be neighbours if there exists a direct link between the nodes.
Let C be the user given concept and \( N[C] \) is closed neighbourhood of concept C. Then, \( N[C] \) is semantically expanded query for the single concept match. Figure 4 represents an example of query expansion for single concept match with the Ontology.

Figure 3. Working of SAIRO engine.

For example ‘Internet’ is the single user given term matches with the Ontology, and all the neighbouring concepts along with the relation type is added to the search term.

Even though the documents with ‘Internet’ keyword is not available, there are some other documents with keyword ‘2G’ and ‘ARPANET’ will be retrieved which will still provide the user to get some information about the ‘Internet’.

Query: Internet

Expanded Query: Internet also called ARPANET
Internet provides e-Mail
Internet uses TCP
2G access to Internet
VOIP uses Internet

2) Concept-Relation Match: If there is a match with single concept and single relation, then the associated concept is also used in document retrieval.

Let X and Y be two different concepts and R be a relation, then R (X, Y) implies X is related to Y via relation R. When user given terms consists of X and R then Y must also be added to the search terms such that R (X, Y).

Figure 5 shows the example of concept relation match, where user term ‘Internet Society’ match with the concept and the term ‘publish’ matches with the corresponding relation in the Ontology. ‘RFC’ is the associated concept for the concept-relation match, which is also included in the search term. This type of semantic query expansion helps to retrieve the appropriate term, as the user search.

Figure 5. Expansion of concept-relation.
3) Concept-Concept Match: If two or more user given terms match with the concepts, all the intermediate concepts associated with the terms are included to the search term.

Let X, Y be two different concepts and F[X, Y] represents the flow between the concepts, then generic representation to expand the user terms is \( \Sigma F[X, Y] \).

Figure 6 shows the generic model of two concepts C1 and C2 and the undirected flow between the concepts, which include all the intermediate concepts that can relate the two main concepts in one or more relations.

IV. IMPLEMENTATION

SAIRO is implemented as web application using Java2EE platform. Protégé 4.3 is used to develop computer network Ontology, which acts as main source to the SAIRO. Eclipse Kepler IDE is used to develop the SAIRO application and its being published in Apache Tomcat 7. HTTP Tracker is used to crawl the entire site. Java search engine library Apache Lucene 3.2 is used to search for documents. The pseudo code for SAIRO engine to expand the user query is shown in figure 7.

```java
if(SingleConceptMatch(userKeyword)==true) 
{  
  query.AddNeighbours(userConcept) 
} 
else if(ConceptRelationMatch(userKeyword)==true) 
{ 
  query.AddAssociatedConcept(userConcept, userRelation) 
} 
else if(ConceptConceptMatch(userKeyword)==true) 
{ 
  query.AddAllFlow(userConcept1, userConcept2) 
}
```

Figure 7. Pseudo code of SAIRO.

SAIRO uses bidirectional breadth first search to retrieve the flow between the two concepts, there by adding all intermediate terms between the concepts from Ontology to the search terms. Stemming is used by SAIRO in order to remove the repeated and less relevant terms after semantically expanding the user query. For example terms like 'is a' is common association which degrade the performance and such terms are removed from search terms. 'SAIRO Search' performs the actual search operation. Identify the concepts and relations from the user input query. The algorithm for SAIRO search is shown in figure 8.

1. If user input match with single concept in Ontology
   a. Fetch all the neighbouring concepts associated with user concept from the Semantic knowledge base (Ontology).
2. Else if user input matches with single concept and relation
   a. Fetch the associated concept for the user entered concept and relation from the Ontology.
3. Else if user input matches with two or more concepts
   a. Retrieve the flow between all the concepts using bidirectional breadth first search.
   b. Add the intermediate concepts in the flow along with the relations to the search query.
4. Retrieve the documents based on the new search term.

Figure 8. Algorithm of SAIRO.
Figure 9 shows the snapshot of SAIRO search being performed with two concepts. In the implementation, 'Do you mean' part contains the semantically expanded terms performed by SAIRO, which is also included in retrieving the documents.

Figure 9. Snapshot of SAIRO search with two concepts.

HTML/ CSS is used in front-end design. JavaScript is used for client side validation. Servlets are used in SAIRO to perform server side manipulations. HttpSession is used to track the user session. Initially, SAIRO parses OWL file to get the concepts and relation between them.

SAIRO can be integrated with most of the existing search engines by initially developing Ontology. As the user query is received, it must be passed to the SAIRO engine, where the query is semantically expanded along with association rules with the help of Ontologies. These expanded queries are used for further processing in the search engine for retrieving the relevant documents.

V. EVALUATION

Information retrieval systems are evaluated using precision and recall measure. Time complexity of SAIRO varies on three main cases. On single concept match, it executes in $O(N(C))$, where $N(C)$ is the open neighbourhood of user given concept $C$. The best case takes a linear time of $O(1)$, in case of concept-relation match, where it. The worst case is concept-concept match, where it takes $O(C+R)$, where $C$ is the total number of concepts and $R$ is the total number of relations in the Ontology.

SAIRO's performance is evaluated by comparing the precision, recall and F-measure with Google, Bing and Lucene for ten sample queries. Figure 10 shows the precision of SAIRO with other search engines. Figure 11 shows the average precision of SAIRO with other search engines. SAIRO has a precision value of 0.78, which is 8% more than Lucene, 16% more than Google and 17% more than Bing.

Figure 10. Comparison of Precision.

Figure 11. Comparison of Mean Precision.
In figure 12, the recall measure of SAIRO is compared with other search engines and figure 13 shows the average recall measure of SAIRO with other search engines. SAIRO got a recall value of 0.53, which is 6% greater than Lucene, 9% more than Google and 12% more than Bing.

Figure 12. Comparison of Recall.

![Figure 12. Comparison of Recall.](image1)

Figure 13. Comparison of Mean Recall.

![Figure 13. Comparison of Mean Recall.](image2)

Figure 14 shows the F-Measure of SAIRO compared with other search engines, where SAIRO have F-Measure value of 0.63, which is 7% more than Lucene, 12% more than Google and 14% more than Bing.

Figure 14. Comparison of F-Measure.

![Figure 14. Comparison of F-Measure.](image3)

Figure 15 shows the number of terms in the input query, expanded query and relevant terms in the expanded query. SAIRO engine got an average of 92.5% relevant terms in the expanded query with 57.5% of new relevant terms than the input terms. Time factor is not compared with Google and Bing, as it runs in different servers and consumes different time based on the hardware of server and intermediate nodes connecting the server and the client.

Figure 16 shows the time taken by SAIRO and Lucene published in same server, where SAIRO consumes an additional time of 0.01 second in average.

Figure 15. Input terms to relevance in expanded terms of SAIRO.

![Figure 15. Input terms to relevance in expanded terms of SAIRO.](image4)
Figure 16. Execution time of SAIRO and Lucene.

SAIRO exceeded the performance of Lucene, Google and Bing results in terms of precision, recall and F-Measure. There is a considerable increase in time factor for performing the query expansion, which helps SAIRO to achieve better precision and recall.

VI. CONCLUSIONS

SAIRO is an information retrieval model, which deeply explores the association between the user terms to retrieve more relevant information. Several models to incorporate semantic technology in information retrieval have been developed, but complete understanding of the relations between the user-entered concepts can make the search engine to understand more about the user context and promote the search from passively retrieving to actively responding. SAIRO proves that an effective query can fetch expected result with its remarkable precision and recall scores having F-measure of 0.63, which is 7% greater than Lucene, 12% greater than Google and 14% greater than Bing.

VII. FUTURE WORK

An extended way of indexing the documents semantically using association rules would improve the search level. Once the documents are crawled, it must be pre-processed for indexing. A new model to store the index semantically by knowing the association between the documents could enhance the search results.

ACKNOWLEDGMENT

The evaluation is performed with Lucene 3.2 version and the comparison with Google and Bing is based on the result obtained on the same site on 15th March 2014.

REFERENCES


